Session 122

Using Cocoa

Ali Ozer
Manager, Cocoa Frameworks

What We Will Do

- Cover some Cocoa programming topics
- Discuss API
- Write code
- Run demos
- Demonstrate that Cocoa is powerful and easy

Quick Overview of Objective-C

- Small superset of C
  - ANSI C
  - Some additional syntax
  - A few additional types
- Dynamic object runtime

Quick Overview of Objective-C

- Defining a method

```
- (void) setWidth: (float)w  height: (float)h {
        width = w;
        height = h;
}
```

- Calling a method

```
[ myRect  setWidth: 42  height: 3.14 ];
```

- Referring to a method

```
SEL mySel = @selector ( setWidth:height: );
```

Quick Overview of Objective-C

- Referring to an object

```
Rectangle *myRect;

id someOtherObject;
```

- Referring to this object

```
[ self  setWidth:42  height:3.14 ];
```

- Constant strings

```
NSString *str = @"Hello World";
```

Today's Talk Format

- You ask the questions
- We answer them
- And we already thought of the questions you want to ask

Question:

I wrote a small app for my school project but for full credit I need to support "undo."

The app is so simple I don't know if I can! Help me!

Undo

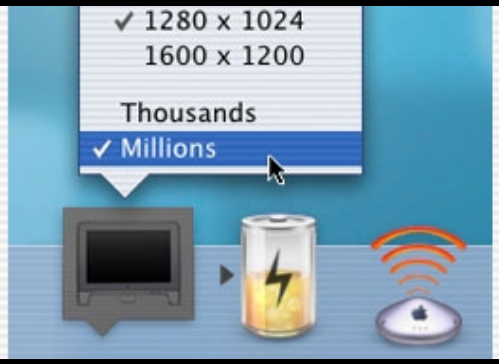
- Undo is easy to add to Model-View-Controller applications
- However, it can easily be added to other apps as well

NSUndoManager

- Usually one per window or document
- On an undoable user action, you register a callback which will be invoked to undo the action

```
- (void) registerUndoWithTarget: (id)callbackTarget
                       selector: (SEL)callbackSelector
                         object: (id)callbackArgument
```

Demo



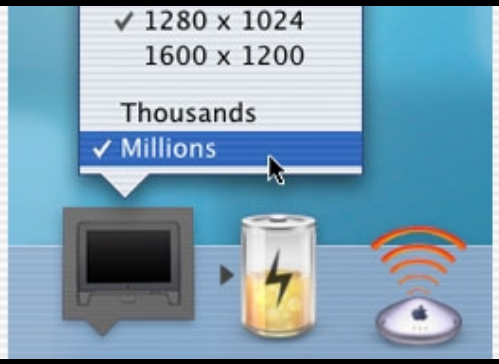
NSUndoManager

- For more sophisticated callbacks, use

```
- (id) prepareWithInvocationTarget: (id)callbackTarget
```

- This method allows you to “freeze-dry” any call, and replay it back later

Demo



Question:

How do I add a toolbar to my application? I don't see it in Interface Builder.

Toolbar

- New in 10.0
- Two classes:
  - NSToolbarItem
  - NSToolbar

NSToolbarItem

- Represents items in the toolbar
- Attributes include image, label, tool tip, menu, target, action
  - Can also have a custom view
- Each item also has a unique identifier (NSString)
  - Used by the toolbar to reference the items
  - Some built-in identifiers provided by Cocoa

NSToolbar

- Represents the whole toolbar
- Attributes include whether it's visible, customizable, etc.
- Uses identifier to save state and remain synchronized with other toolbars
- Relies on its delegate for information about items

NSToolbar Delegate Methods

- Return all the allowed toolbar item identifiers

```
- (NSArray *) toolbarAllowedItemIdentifiers: (NSToolbar *)bar
```

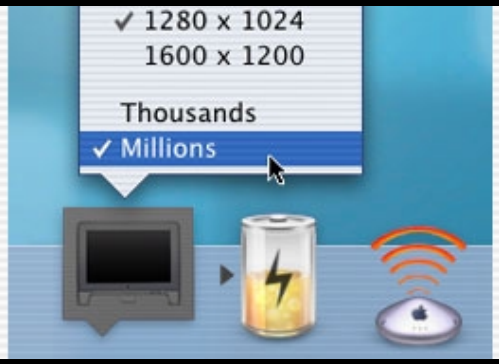
- Return the initial set of identifiers

```
- (NSArray *) toolbarDefaultItemIdentifiers: (NSToolbar *)bar
```

- Create a toolbar item

```
- (NSToolbarItem *) toolbar: (NSToolbar *)bar
                     itemForItemIdentifier: (NSString *)ident
                     willBeInsertedIntoToolbar: (BOOL)flag
```

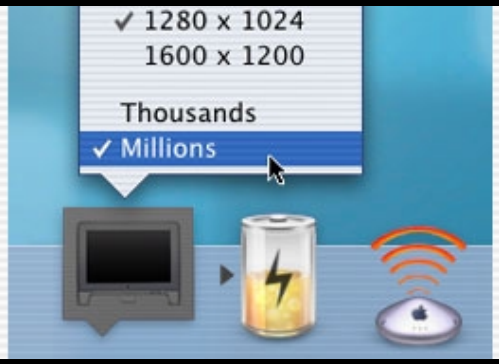
Demo



Question:

How can I get a better frame rate out of my animating game?

Demo



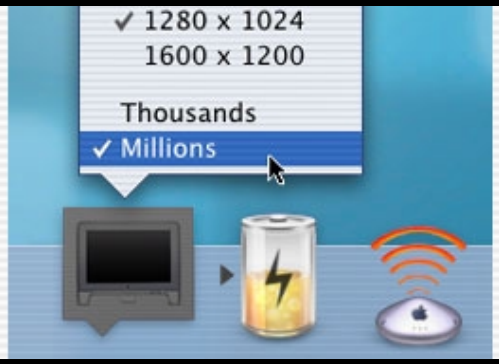
Drawing

- Draw as little as possible
  - Fine-tune the update area
  - Use isOpaque to prevent superviews from being drawn needlessly
- Use QuartzDebug to see what's drawing

Question:

Hmm, interesting. But it's still not good enough for my boss. Can I make it draw even faster?

Demo



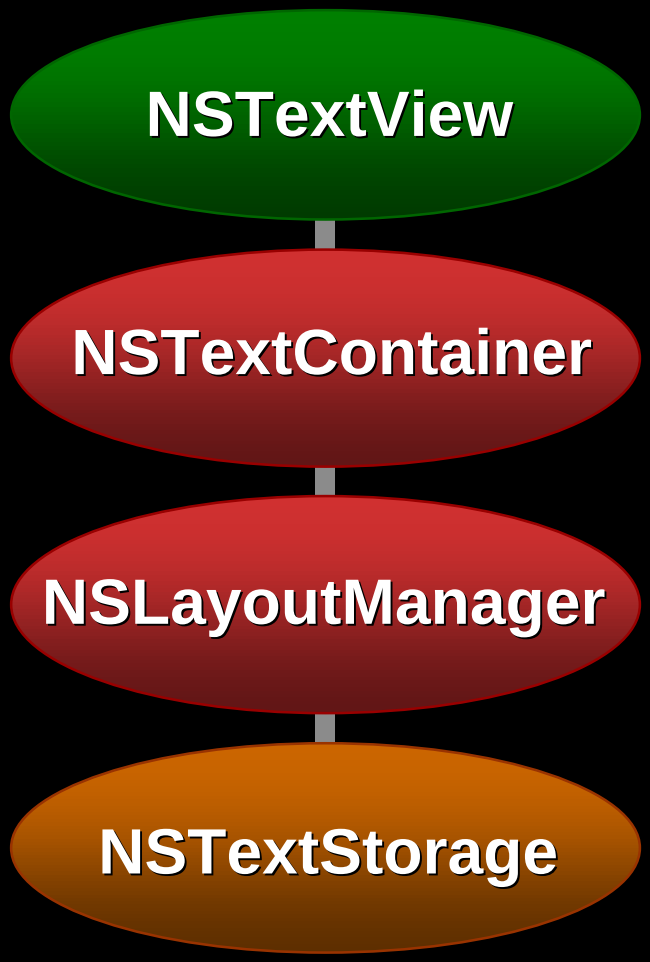
String Drawing

- We use the following NSString API:

```
- (void) drawInRect: (NSRect)rect
                 withAttributes: (NSDictionary *)attrs
```

- This method is convenient, but does a lot of work under the covers
- We can try using the text system directly for some performance gain

Text System Architecture



Handles display and editing

Provides text area info

Computes and stores glyphs and their locations

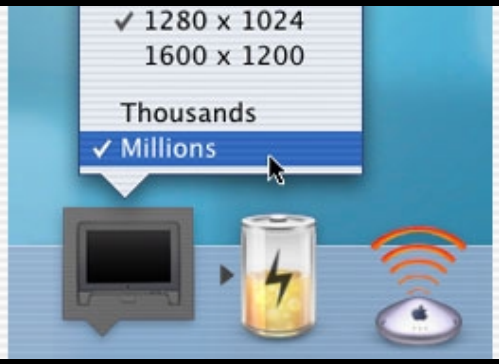
Stores characters and their attributes

NSLayoutManager

- Gets text contents from NSTextStorage
- Converts characters in text to “glyphs”
- Computes glyph locations
- Stores this info and draws the glyphs when asked by the view

```
- (void)drawGlyphsForGlyphRange: (NSRange)range
                        atPoint: (NSPoint)point
```

Demo

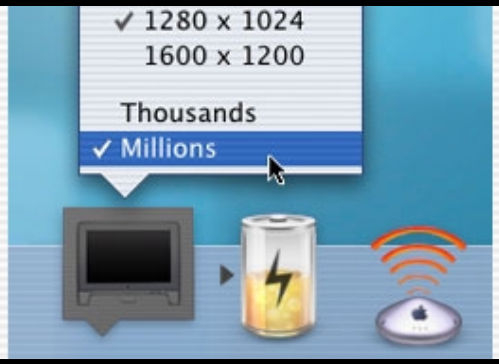


Question:

Hey, that's sweet! Now any ideas on how I can add some pizzazz to the app?

Oh, we're shipping the app this afternoon, so something quick, please.

Demo



NSColor

- Represents not only colors, but “meta” colors, “catalog” colors, patterns, . . .
- To paint with an NSColor, just do [NSColor set]
- To create an NSColor, use methods such as:

```
+ (NSColor *) colorWithPatternImage: (NSImage *)image
+ (NSColor *) whiteColor
+ (NSColor *) colorWithCalibratedRed: (float)r
                green: (float)g blue: (float)b alpha: (float)a
+ (NSColor *) windowBackgroundColor
```

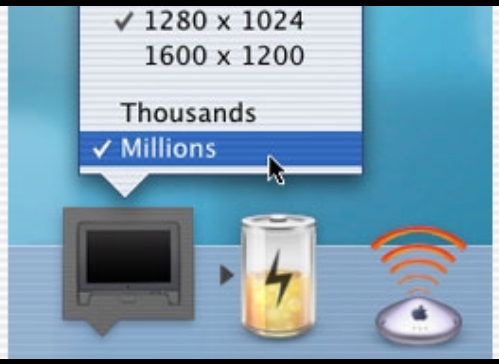
Question:

So, how does the magic typing assistant work?

Services

- Services allow applications to provide their functionality to other applications
- The magic code typing assistant service:
  - Is provided by a modified TextEdit app
  - The produced text is simply the next line's worth of text from the current document in TextEdit; cursor is also advanced a line

Demo



What We Did Today

- Added undo to an unsuspecting app
- Added a customizable toolbar to TextEdit
- Made a game draw much faster
- Learned how to use patterns to spiffy up any app
- Saw how useful services can be

Conclusion

- We saw how to enhance apps in user-visible ways with just a little bit of code
- And do some sophisticated things with a bit more code

Cocoa: Simple things simple,
Complex things possible

Roadmap

102 Cocoa Overview	Hall 2 **Mon., 5:00pm**
702 Interface Builder	Hall 2 **Tues., 3:30pm**
123 Advanced Cocoa Topics	Hall 2 **Thurs., 5:00pm**
705 Apple Performance Tools	Room A2 **Thurs., 5:00pm**
FF112 Feedback Forum: Cocoa	Room A2 **Fri., 5:00pm**

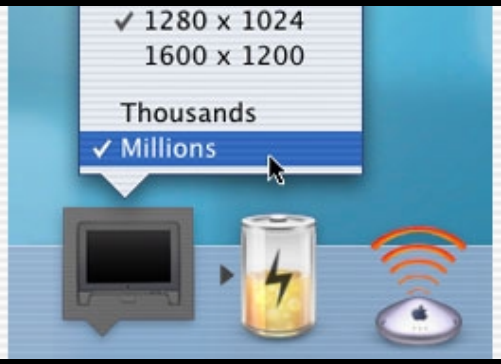
Who to Contact

Heather Hickman
Cocoa Technology Manager
Apple Worldwide Developer Relations
hhickman@apple.com

Cocoa Feedback
cocoa-feedback@apple.com

Cocoa Development Mailing List
Subscribe at
www.lists.apple.com/mailman/listinfo/cocoa-dev

Q&A



Heather Hickman
Cocoa Technology Manager
hhickman@apple.com

http://developer.apple.com/events/urls.html

WWDC
Worldwide Developers Conference 2001